CLAIMS

What is claimed is:

Sub A2

1. A multi-processor computer system, comprising:

a plurality of processors, each processor coupled to at least one memory cache, one cache control unit, and one interprocessor router;

a memory coupled to each processor, each memory managed by a memory controller configured to accept memory requests from the plurality of processors; and

at least one input/output device coupled to at least one processor;

wherein the memory requests from a local processor are delivered to the memory controller by the cache control unit and wherein memory requests from other processors are delivered to the memory controller by the interprocessor router and wherein the memory controller allocates the memory requests in a shared buffer using a credit-based allocation scheme .

2. The computer system of claim 1 wherein:

the cache control unit and the interprocessor router are each assigned a number of credits;

at least one of said credits must be delivered by the cache control unit to the memory controller when a memory request is delivered by the cache control unit to the memory controller; and

at least one of said credits must be delivered by the interprocessor router to the memory controller when a memory request is delivered by the interprocessor router to the memory controller;

wherein if the number of filled spaces in the shared buffer is below a threshold, the buffer returns the credits to the source from which the credit and memory request arrived.

3. The computer system of claim 2 wherein:

wherein if the number of filled spaces in the shared buffer is above a threshold, the buffer holds the credits and returns a credit in a round-robin manner to a source from which a credit has been received only when a space in the shared buffer becomes free; and

wherein if a source has no available credits, that source cannot deliver a memory request to the shared buffer.

4. The computer system of claim 2 wherein:

the number of credits assigned to the cache control unit and the interprocessor router is sufficient to enable each source to deliver an uninterrupted burst of memory requests to the buffer without having to wait for credits to return from the buffer.

5. The computer system of claim 4 wherein:

the number of credits available in the cache control unit and the interprocessor router are stored and updated in counters located in the cache control unit and the interprocessor router; and

the number of credits spent by the cache control unit and the interprocessor router are stored and updated in counters located in the shared buffer.

6. The computer system of claim 4 wherein:

the threshold is the point when the number of free spaces available in the buffer is equal to the total number of credits assigned to the cache control unit and the interprocessor router.

7. A computer processor for use in a multi-processor system, comprising:

an associated memory;

a memory controller comprising a request buffer in a front-end directory in-flight table;

an L2 data cache;

an L2 instruction and data cache control unit configured to send request and response commands from the processor to the memory controller;

at least one input/output device coupled to the processor; and

an interprocessor and I/O router unit configured to send request and response commands from other processors to the memory controller;

wherein the L2 instruction and data cache control unit and interprocessor and I/O router unit are assigned a number of credits and are configured to give up a credit to the directory in-flight table each time a request or response command is sent to the request buffer and wherein if the request buffer is filled below a buffer threshold, the directory in-flight table immediately returns credits to the source from which the credit was received.

8. The computer processor of claim 7 wherein:

if the request buffer is filled above a buffer threshold, the directory in-flight table holds credits and returns a credit to a source from which a credit was received only when a buffer space is emptied; and

wherein if a source has no available credits, that source may not send a request or response command to the request buffer and wherein if a source has one available credits, that source may only send a response command to the request buffer.

9. The computer processor of claim 8 wherein:

the credits are returned to the sources which have given up credits to the directory in-flight table in a random, equally probably manner.

10. The computer processor of claim 8 wherein:

the buffer threshold is the point above which the number of empty spaces in the request buffer is equal to the total number of credits assigned to the L2 instruction and data cache control unit and interprocessor and I/O router.

11. The computer processor of claim 8 wherein the directory in-flight table further comprises:

a counter to store and update the number of credits spent by the L2 instruction and data cache control unit;

a counter to store and update the number of credits spent by the interprocessor and I/O router; and

a counter to store and update the number of empty spaces in the request buffer when the request buffer is filled above the buffer threshold;

wherein when the request buffer is filled above the buffer threshold, the directory in-flight table holds credits and returns credits only when the number of empty spaces in the buffer increases.

12. The computer processor of claim 8 wherein:

the number of credits available to the L2 instruction and data cache control unit and interprocessor and I/O router is stored and updated by counters in each unit.

13. The computer processor of claim 8 wherein:

the number of credits available to the L2 instruction and data cache control unit and interprocessor and I/O router is determined by the round trip time required to send a credit to and receive a credit from the directory in-flight table;

wherein the number of credits given to each source is sufficient to allow each source to send an uninterrupted sequence of request or response commands to the directory in-flight table without delays caused by waiting for credits to return from the directory in-flight table.

14. A method of allocating space in a shared buffer, comprising:

assigning credits to each source that sends data packets to the shared buffer; and

requiring each source to spend a credit each time that source sends a data packet to the shared buffer;

wherein if the number of empty buffer spaces is larger than a buffer threshold, immediately paying the credit back to the source from which the credit and data were sent; and

wherein if the number of empty buffer spaces is smaller than the buffer threshold, holding the credit until a buffer space becomes empty and then paying a credit back to a source from which a credit was sent.

15. The method of claim 14, wherein:

when the number of empty buffer spaces is smaller than the buffer threshold and a buffer space becomes empty, returning a credit in a random, equally probably manner to one of the sources which have spent credits held by the buffer.

16. The method of claim 14, wherein:

when the number of empty buffer spaces is smaller than the buffer threshold and a buffer space becomes empty, returning a credit in a random, statistically skewed manner to one of the sources which have spent credits held by the buffer.

17. The method of claim 14, further comprising:

assigning a minimum number of credits to each source that is sufficient to allow each source to send a continuous sequence of data packets without waiting for returned credits.

18. The method of claim 14, further comprising:

preventing a source from delivering a data packet to the shared buffer if that source has no available credits.

19. The method of claim 14, further comprising:

setting the buffer threshold equal to the number of total credits assigned to all the sources.

20. The method of claim 14, further comprising:

using a counter in each source and a counter for each source coupled to the buffer to track spent and paid back credits.